American Journal of Science, Engineering and Technology
2024; 9(1): 1-13
http://www.sciencepublishinggroup.com/j/ajset
doi: 10.11648/j.ajset.20240901.11
ISSN: 2578-8345 (Print); ISSN: 2578-8353 (Online)

A Comparative Study on the Application of Intelligent Models in the Estimation of Backbreak in Mine Blasting Operations

Festus Kunkyin-Saadaari*, Victor Kwaku Agadzie, Richard Gyebuni

Department of Mining Engineering, University of Mines and Technology, Tarkwa, Ghana

Email address:

fsaadaari@umat.edu.gh (Festus Kunkyin-Saadaari), andersonkwaku538@gmail.com (Victor Kwaku Agadzie), rgyebuni@umat.edu.gh (Richard Gyebuni)

*Corresponding author

To cite this article:

Festus Kunkyin-Saadaari, Victor Kwaku Agadzie, Richard Gyebuni. (2024). A Comparative Study on the Application of Intelligent Models in the Estimation of Backbreak in Mine Blasting Operations. *American Journal of Science, Engineering and Technology*, *9*(1), 1-13. https://doi.org/10.11648/j.ajset.20240901.11

Received: December 7, 2023; **Accepted**: January 4, 2024; **Published**: January 18, 2024

Abstract: Backbreak in the mining industry presents a considerable challenge, impacting both safety and operational efficiency. Accurate prediction of backbreak is therefore a critical endeavour. This study rigorously evaluates four advanced machine learning (ML) techniques—Lagrangian Support Vector Machine (LSVM), Radial Basis Function Neural Network (RBFNN), Gaussian Process Regression (GPR), and Extreme Gradient Boosting (XGBoost)—to ascertain the most effective method for backbreak prediction. Utilising a comprehensive dataset of 60 blasting rounds from the Damang Goldfields Open Pit Mine and prior to the analysis, this dataset underwent a thorough preprocessing phase. The efficacy of each model is assessed using a suite of metrics, including correlation coefficient (r), coefficient of determination (R^2), mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE). The performance of the models is quantitatively compared, revealing XGBoost as the superior predictor in this context, characterised by an r of 0.9788, an R^2 of 0.9565, an MSE of 0.1714, an RMSE of 0.4139, and an MAE of 0.2819. The findings of this study underscore the potential of XGBoost as a robust tool for backbreak prediction, offering mining companies a viable solution to enhance safety protocols and mitigate financial losses related to backbreak incidents. This research contributes significantly to the field of predictive analytics in mining, providing a comprehensive comparative analysis of various ML techniques for backbreak prediction.

Keywords: Backbreak, Blasting, Machine Learning (ML), Cosine Amplitude Method (CAM), Simple Linear Regression (SLR)

1. Introduction

Backbreak is a prevalent issue in rock blasting operations that can result in safety risks and decreased production [1]. In the mining industry, "backbreak" refers to shattered rocks extending behind the last row of holes or past the planned excavation limit in a blast pattern [2]. Bhandari [3] asserts that the breakage and fractures are mostly caused by the incorrect use of explosive energy in blastholes. When an explosive charge within a blasthole detonates, high-pressure shockwaves, heat, and gases are generated rapidly [4]. The walls of blastholes are subjected to the generated gas pressure, which exerts significant stress on the surrounding rock mass (media). Under these conditions, rock breaking occurs when the free surface is sufficiently near the blasthole. If the energy created during blasting is not contained, undesirable traits like backbreak become evident [5]. A study conducted by Konya et al. [6] identified that the key factors contributing to backbreak include an excessive burden, extremely rigid benches, deep stemming on stiff benches and an inappropriate timing delay. According to Gates et al. [7], the main causes of backbreak are insufficient delay time and increased blasting rows. This can result in various effects, including equipment dropping, inadequate fragmentation, unstable mine walls and decreased drilling effectiveness [8]. Accurate prediction of these effects is essential for their

mitigation and the development of open-pit mining in a safe and sustainable manner.

Numerous studies have been conducted to determine the elements that influence the severity of backbreak. These elements can be broadly classified into controllable and uncontrolled factors. While the uncontrollable factors such geological discontinuities are intrinsic and cannot be changed, the controllable factors relate to the parameters that can be altered. Some controlled parameters are specific charge, blasthole diameter, blasthole depth, burden, bench height, spacing and stemming whereas rock mass properties and structural terrain are uncontrollable parameter. In the past, empirical models such as Holmberg Persson, Langefors–Kihlstrom, General Predictor and Gupta [9-11] for blast pattern design were created to achieve objectives such as appropriate fragmentation, minimising blast-induced ground vibration, decreasing backbreak, reducing flyrock throw distance and reducing boulders. However, using such algorithms to predict backbreak is difficult. Additionally, the empirical models only consider a portion of the key blasting operating factors. There are several associated parameters with unclear interrelations, which adds to the complexity of such challenges. The effectiveness of earlier proposed empirical models of backbreak is poor [12]. Unquestionably, this has contributed to the creation of several predictions.

Recently, new predictive models have been developed by numerous researchers for predicting backbreak. The methods encompass a range of techniques such as adaptive neuro-fuzzy inference system [13], hybrid artificial neural network and bee colony algorithm [14], rock engineering system, stochastic modelling [15], fuzzy set theory, support vector machine [16], artificial neural networks and genetic programming [17]. Each of these predictive algorithms has different benefits and drawbacks.

The objective of this research is to develop and compare the effectiveness of four distinct machine learning (ML) models for predicting backbreak in rock blasting: Lagrangian Support Vector Machines (LSVM), Radial Basis Function Neural Networks (RBFNN), Gaussian Process Regression (GPR) and Extreme Gradient Boosting (XGBoost). These models have emerged as a promising alternative for predicting backbreak. These simulations have shown excellent potential for successfully forecasting backbreak during rock blasting, given their outstanding performance compared to other newly proposed predictive models by various researchers such as Yu et al. [16], Nabavi et al. [18] and Arthur et al. [19]. Each model will be assessed to determine its effectiveness and appropriateness for real-world application. The development of an algorithm for each model will be required for this evaluation. The findings of this study will give helpful information about which of the four models is highly recommended for accurate backbreak prediction. Thus, using LSVM, RBFNN, GPR and XGBoost models for predicting backbreak in rock blasting operations is anticipated to provide significant benefits in effectiveness, safety, and environmental conservation and aid in developing more efficient blasting methodologies.

2. Study Area and Data Description

2.1. Study Area Description

Damang mine, a large-scale open-pit gold mining site, is situated in the Western Region of Ghana, specifically within the prolific Tarkwaian gold belt known for its abundance of gold resources. Located approximately 10 km north of the key town of Tarkwa, the mine falls within the gold-rich Paleoproterozoic Birimian Supergroup on the West African Craton. It is a registered company in Ghana with a land area of about 8,111 hectares, including five Prospecting Licenses and two Mining Leases. The mine area, characterised by its undulating hills and valleys, ranges from 100 to 500 m above sea level. Gold mineralisation within the mine is associated with shear zones and quartz veins intersecting the region's prevalent metavolcanic and metasedimentary rocks, namely mafic volcanic rocks and granitoids. To reveal the gold-bearing areas, soil and rock are carefully removed before the valuable metal is extracted from the ore through drilling, blasting, loading, hauling, crushing, grinding, and chemical extraction processes. The Damang mine processing facility handles approximately 5.2 million tonnes of ore annually, with around 5 % oxide and 95 % fresh ore. The introduction of a secondary crushing plant in April 2010 further enhanced the efficiency of the extraction process. Ore is recovered using a combination of open-pit mining operations and existing surface stockpiles, which resulted in a total processed feed of 2.5 million tonnes in the six months ending December 31, 2010, resulting in an output of 117,000 ounces of gold with a grade of 1.46 g/t. In line with high environmental and safety regulations, the mine is committed to sustainable mining practices, including reducing soil erosion, managing waste rock and tailings, reducing dust emissions, and performing reclamation and restoration programmes to restore damaged land areas post-mining to profitable use.

Figure 1. *Location of Damang Mine.*

2.2. Dataset Description

The aim of this study is to create intelligent models using LSVM, RBNFF, GPR and XGBoost to predict backbreak at Damang mine and determine the most effective predictor. To accomplish this objective, a dataset comprising 60 blasting rounds conducted at Damang mine was compiled. The input parameters used for predicting backbreak include specific charge [powder factor] (P), geometric stiffness ratio (G), stemming height (T), burden (B), and spacing (S). These values were obtained from the blasting designs, and other parameters such as the number of blast holes, charge in one hole [cooperating charge] in kg, depth of the blasthole (m) and specific charge [powder factor] were calculated. All of these factors were recorded and compiled in Table 1 as the main controllable blasting factors. In partitioning the data for training and testing, cognisance was taken of the fact that to generate accurate predictions, machine learning (ML) technique requires a sufficient amount of training data [20]. However, when there is an excess of training data, the model may overfit, leading to poor performance when dealing with new data [21].

One important point to note is that the method used to divide the dataset into training and validating sets, either cross-validation or the percentage split approach can have an impact on the model's performance [22]. If the training set is too limited, the model may not have adequate data to learn from and may perform inadequately [23]. Conversely, if the testing set is too small, it may not be a reliable representation of the entire dataset, and the model's performance may not be assessed precisely. While there is no consensus on the optimal ratio for dividing the data, the percentage split approach is a commonly used and successful method for machine learning (ML) modelling [24]. As a result, this technique was chosen for the current data partitioning. The training set included 42 data points, representing 70 % of the entire data, and was utilised to create and train various models. In contrast, the testing set consisted of 18 data points, accounting for 30 % of the data, and was used to evaluate the models' performance with new and unseen data.

Table 1. *A Statistical Analysis of the Parameters Collected from the Damang Mine.*

Parameters	Unit	Min	Max	Mean	Std Dev
Powder factor (P)	kg/m^3	0.15	0.90	0.457	0.208
Geometric stiffness (K)	--	2.40	6.93	3.311	0.791
Stemming height (T)	m	1.80	4.50	3.599	0.679
Spacing (S)	m	2.74	6.00	4.639	0.696
Burden (B)	m	2.00	6.50	3.617	0.794
Backbreak	m	1.00	10.0	4.124	2.091

3. Methods Used

The machine learning (ML) techniques employed to develop models are discussed in this section.

3.1. Lagrangian Support Vector Machine (LSVM)

According to research by Behzad et al. [25] and Vapnik [26], the Support Vector Machine (SVM) was developed in the 1990s as a practical solution for classification and regression applications. Because of its ability to efficiently learn from a minimal set of parameters, it has been successfully applied in many different fields such as stock price forecasting [27], time series analysis [27], and Flyrock prediction [28]. SVMs have shown effectiveness in regression tasks as well as classification tasks by reducing structural risk, as noted in studies by Mukherjee et al. [29] and Jeng et al. [30]. In order to approximate the value of *y(x)* based on the supplied data, it uses a method that comprises determining a function that is similar to *f(x) as shown in Equation (1).*

$$\text{The given data set } (x_1, y_1) \ldots, (x_m, y_m) \text{ belongs to } X \subseteq R^n \text{ and } Y \subseteq R \quad (1)$$

The function is predicted using support vectors, a subset of training data. The SVR method also generates a sparse characteristic using ε -insensitive loss function [31]. Mathematically, the definition, this function is is presented in Equation (2).

$$|19 - f(x)|_\varepsilon = \begin{cases} 0 \ if \ |_{y-f(x)\leq}\varepsilon| \\ |y - f(x)| - \varepsilon \ otherwise \end{cases} \quad (2)$$

In the aforementioned equation, the predicted value of *y* is denoted by the function *f(x)*, and any errors less than the ε - limit are not penalised. The regression method comprises using a linear function approximation, and the input vector *x* is stated as follows [32, 33]:

$$f(x) = (w.x) + b \text{ where } w, x \in X \subseteq R^n, b \in R \quad (3)$$

In this context, the bracket notation is used to represent the inner product of two vectors within a Hilbert space, which is a unique type of vector space. This space is characterized by an inner product that specifies a distance function, thus making the space a complete metric space [34]. The regulated risk functional (*Rreg*) has to be defined in order to assess *f(x)*. Equation (4) presents a description of *Rreg*:

$$R_{reg}[f] = \frac{1}{2}\|w\|^2 + CR_{emp}[f] \text{ where } R^{\varepsilon}{}_{emp}[f] = \frac{1}{m}\sum_{i=1}^{m}|y_i - f(x_i)|\ \varepsilon \quad (4)$$

The following convex and limited quadratic optimisation issue can be solved in a manner similar to reducing the value of Equation (4):

$$L(w, \xi, \xi') = \frac{1}{2}\|w\|^2 + c\sum_{i=1}^{N}(\xi_i + \xi') \quad (5)$$

$$\text{Subject to } \begin{cases} y_i - w^T X - b \leq \xi + \varepsilon \\ \omega_T x + b - y_i < \xi' + \varepsilon \\ \xi_i, \xi'_{x_i} \ and \ x_i \geq 0 \end{cases}$$

In Equation (5), the parameter *C* = "capacity" is employed to ensure the margin ε is maximised while minimising the classification error ξ. A larger value of *C* suggests that greater weight is placed on misclassifications in the training set, which lowers the machine's capacity for generalisation. Low

generalisation ability can cause a machine to perform well on the training set but poorly on entirely new, untested data. When the machine overfits the training data, poor generalisation may result, especially if the training data contains unusual and irregular patterns. On the other hand, choosing a smaller value of *C* lessens the chance that the Support Vector Machine (SVM) may become overfit to the training set of data. Equation (5) also states that any error smaller than ε does not need a nonzero value for either ξ_i or ξ'_i because it does not contribute to the objective function [35].

By allowing parameters *C* and ε to have non-zero values, the optimal equation for the hyperplane can be obtained by maximising the following equations through the introduction of Lagrange multipliers (α, and α'):

$$\mathrm{L}(\alpha,\alpha')=\frac{1}{2}\sum_{i=1}^{N}\left(\sum_{i=1}^{N}(\alpha-\alpha')x_i'x_i(a_i-a_i')+\sum_{i=1}^{N}\left((a_i-a_i')y_i-(a_i+a_i')\varepsilon\right)\right. \quad (6)$$

Subject to $0 \le (a_i - a_i') \le c$

In order to improve generalisation in the non-linear condition, data points are mapped onto feature space:

$$x_i x_j \rightarrow \varphi(x_i)\varphi(x_j) \quad (7)$$

Exact value of the function $\varphi(x_i)$, is not required since it can be determined by choosing an appropriate kernel function, such that $k\ (x_i x_j) = \varphi(x_i)\varphi(x_j)$. Even in cases where the original input space is nonlinear, it is still possible to separate data in Hilbert space by selecting the correct kernel function. In contrast to the input space, where a hyper-plane is not used to split the data for *n*-parity, this allows the feature space to separate the data using an appropriate kernel [36]. For regression analysis, some of the frequently used kernels are polynomial, sigmoid kernel, radial basis function (Gaussian), laplace RBF kernel and Anove RBF kernels.

According to the definition of the kernel, the nonlinear regression estimation problem of SVR might take the following forms:

$$y_i = \sum_{i=j}^{N}\sum_{j=1}^{N}(\alpha-a_i')\varphi(x_i)^T\varphi(x_j)+b = \sum_{i=j}^{N}\sum_{j=1}^{N}(\alpha-a_i')\,\mathrm{K}(x_i'x_j)+\mathrm{b} \quad (8)$$

Equation (5)'s limitations change to $\xi_i = 0$ if $0 < a_i <$ C and $\xi'_i = 0$ if $0 < a_i' <$ C, which can be used to find *b* [37].

It is well known that a certain collection of parameters, including the capacity parameter *C*, the value of the -insensitive loss function, the kernel *k*, and associated parameters, have a significant role in the effectiveness of SVM. During the training phase, the parameters *C* and ε were given the values 1 and 0.001, respectively. Parameter *C* functions as a regularisation parameter that governs the balance between maximum generalisation and minimum training error. It would be difficult to adequately fit the training data if *C* was set to an extremely small value.

When selecting the value of the loss function (ε) for SVM training, it is crucial to consider the possibility of various types of noise that may be present in the database. The kernel function is another important parameter that needs to be accurately chosen. Studies have shown that the Polynomial kernel function performs better than other kinds of kernel functions [38, 39]. Encouragingly, the coefficient of determination the model significantly increased from 0.7024 to 0.8928 when the polynomial kernel was used instead of the linear kernel. The Polynomial kernel function is represented by the following equation:

$$k(X_i, X_j) = (X_i^T, X_j + 1)^P \quad (9)$$

3.2. Radial Basis Function Neural Network (RBFNN)

A Radial Basis Function Neural Network (RBFNN), a type of multi-layer feedforward artificial neural network, leverages radial basis functions as its activation mechanisms [40]. Such networks are highly effective for tackling both regression and classification tasks [41]. Initially developed within a statistical framework and rooted in function approximation theory, RBFNNs have since been widely applied in machine learning and pattern recognition [42]. The structure of RBFNNs comprises three layers: an input layer, a hidden layer, and an output layer [43]. The input layer's neurons straightforwardly relay the input features to the hidden layer without any alteration [44]. The network's architecture includes a single hidden layer, where each neuron employs a Radial Basis Function, often a Gaussian function, as its activation function (as indicated in equation 10). Each hidden layer neuron acts as a "prototype", synonymous with the center of the RBF. The RBF's output is determined by the distance from the input to this prototype [45]. Lastly, the output layer performs a weighted summation of the inputs it gets from the hidden layer, suggestive of a linear regression operation [46]. RBF networks are often used in function approximation, time series prediction, control, and signal processing, among other applications [47]. The Gaussian function given by Park et al. [48]:

$$\varphi(\|x-c_j\|) = exp\left(-\frac{1}{2\sigma_j}\|x-c_j\|^2\right) \quad (10)$$

Where $\|x-c_j\|$ is the Euclidean distance between x and c_j, x is the input vector, c_j is the center vector and σ_j is the width (expansion) of the Gaussian function. The output of the RBFNN, represented by *y*, can be computed using the formula given by:

$$y = \sum_{i=1}^{j}\varphi(\|x-c_j\|)w_{ij} \quad (11)$$

Where *j* is the number of hidden neurons, x is the input vector, c_j are the centres associated with the hidden neurons, w_{ij} are the weights associated with the hidden neurons, φ is the radial basis function, typically a Gaussian function, and $\|x-c_j\|$ is the Euclidean distance from the input vector to the centre [49]. Figure 2 shows the RBFNN's (radial basis

function neural network) structure.

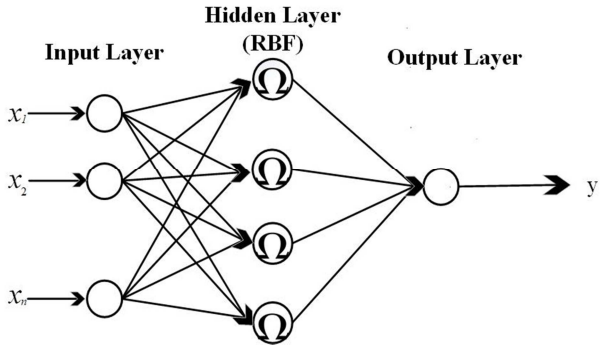

Figure 2. *Network arrangement based on RBF.*

3.3. Gaussian Process Regression (GPR)

A Gaussian process (GP) is a random process, characterized by variables indexed by time or space, and uniquely distinguished by the property that any finite subset of these variables adheres to a multivariate Gaussian distribution [50]. They have been utilized as an effective tool in machine learning, predominantly in the fields of regression and smu classification [19, 51]. Originally, Gaussian Process Regression (GPR) was employed in the forecasting of time series, first introduced by Wiener and Kolmogorov in the 1940s. Subsequently, it gained substantial traction in the field of geostatistics [52], where it's recognized as kriging. A Gaussian Process (GP), represented as *t (x)*, is a stochastic process that is completely defined by a covariance function, also known as a kernel, indicated as *k (x, x')*, and a mean function, signified as *m (x) [50, 53, 54]*. Here, Eqns. (12) and (13) represent these functions:

$$m(x) = E\big(t(x)\big) \tag{12}$$

$$\mathrm{Cov}\big(t(x)\big),t(x') = k(x,x';\theta) = E\left(\big(t(x)-m(x)\big)\big(t(x')-m(x')\big)\right) \tag{13}$$

where θ stands for the collection of hyperparameters. Thus, in Eqn. (14), a Gaussian process is written as follows:

$$t(x)\sim GP\big(m(x),k(x,x')\big) \tag{14}$$

where the gaussian process is denoted by *GP*. This indicates that the function $t\,(x)$ has a Gaussian distribution with a mean of $m\,(x)$ and a correlation function of $k(x,x')$.

Gaussian Process for Regression

Establishing a model that illustrates the link between a response variable, *y*, and a group of predictor variables, x_i, is the goal of every regression work. Any regression function, *t (x)*, may be used to link a response variable, *y*, by using an additive independent Gaussian noise, ε, which generalize the noise in the data. This connection is demonstrated by the equation below:

$$y = t(x) + \varepsilon \tag{15}$$

The Gaussian process represented by Eqn (14) is modified to Eqn (16) with the presence of noise, ε, which has an average of zero and a variance of σ_n^2, denoted as $\varepsilon\sim N\,(0,\,\sigma_n^2)$.

$$t(x)\sim GP(m(x),k(x,x')+\sigma_n^2 I) \tag{16}$$

where the identity matrix is I. According to Eq. (17), which is based on the marginalization property of generalized projections (GPs) and the additive nature of noise, the joint distribution of the test outputs f_* at test points X and the training output y at locations X_* is supplied. According to [55].

$$\begin{bmatrix} y \\ f_* \end{bmatrix} \sim N\left\{\begin{bmatrix} m(X) \\ m(X_*) \end{bmatrix}\begin{bmatrix} k(X,X+\sigma_n^2 I) & k(X,X_*) \\ k(X_*,X) & k(X_*,X_*) \end{bmatrix}\right\} \tag{17}$$

To determine the predictive distribution, the combined Gaussian prior distribution can be conditioned on X, y, and X_* as shown in the provided equation:

$$P(y_*|x,y,x)\sim N\big(\bar{y}_*, var(y_*)\big) \tag{18}$$

The equation shown above includes $\bar{y}_*$ (Eq.19), which denotes the predictive mean, and *var (y∗)* (Eq. 20), which represents the predictive variance. These equations were introduced by Li *et al*. [55].

$$\bar{y}_* = m(x_*) + [k(X_*,X)K(X,X) + \sigma_n^2 I]^{-1}(\text{y-m(X)}) \tag{19}$$

$$var(y_*) = k(X_*,X_*)[K(X,X) + \sigma_n^2 I]^{-1}\, k(X,X_*) \tag{20}$$

Covariance function

It is beneficial to describe how two or more random variables change simultaneously, or to assess the connection between the variables, when they are represented on a probability space [56]. The covariance is a frequently used indicator of the correlation between two random variables [57]. A Gaussian process regression model's core component is the correlation function, and choosing the right covariance function is essential to defining the sample function that will be represented [58]. Its goal values are probably comparable when input points are connected nearby. Consequently, a goal value for a test point close to a training point should also be close to the training point. The constant, linear, Gaussian noise, Ornstein-Uhlenbeck, squared exponential, gamma exponential, Matérn Class, periodic, rational quadratic, and others are examples of frequent covariance functions that may be found in the literature. The squared exponential function is frequently used and stands out in the literature [59].

Training a Gaussian process regression model

The behaviour of the GPR model is governed by the mean function and covariance (kernel) function values, which are the hyperparameters of the Gaussian process. All of the hyperparameters related to the mean and covariance functions must be learnt in order to train and build a GPR model. Methods like sampling or optimisation can be used to accomplish this. However, it is common to use a strategy of maximising the log marginal probability (Eqn. 21), as suggested by Liu *et al*. [60]:

$$log\ \mathrm{p(y|X,\theta)} = -\frac{1}{2}y^T(k+\sigma_n^2 I)^{-1}\mathrm{y}-\frac{1}{2}log\ |k+\sigma_n^2 I| - \frac{n}{2}log 2\pi \quad (21)$$

where θ is a vector made up of all the hyperparameters and y^T is the inversion of the *y* vector.

Moore et al. [61] suggest that the conjugate gradient technique is a potent optimisation algorithm that is gradient-based and can be employed to maximise the log marginal likelihood.

Model Development

The pre-processing stage of the model building involved normalising the training and testing data sets. The purpose of normalisation is to ensure that the variability is consistent and minimise the impact of variables with high variance on the model's predicted outcomes. Since the data sets contain various ranges of values and physical units, Equation. (22) [62] was used to normalise the data into the range of [1, 1]:

$$M_i = M_{min} + \frac{(M_{max}-M_{min})\times(N_i-N_{min})}{N_{max}-N_{min}} \quad (22)$$

where M_{min} and M_{max} are set to -1 and 1, respectively, M_i is the normalised data, N_i is the measured blast data, and N_{max} and N_{min} are the maximum and minimum values of the measured blast data.

In this equation, M_{min} and M_{max} correspond to -1 and 1, respectively. M_i represents the normalised data, whereas N_i represents the measured blast data. N_{max} and N_{min} denote the highest and lowest values of the measured blast data.

3.4. Extreme Gradient Boosting (XGBoost)

XGBoost, short for eXtreme Gradient Boosting, is an enhanced and scalable rendition of the gradient boosting technique. It has been specifically developed to enhance efficiency, computational speed, and overall model performance [63]. XGBoost builds an incremental extension of the objective function by the reduction of a loss function, much as gradient boosting [64]. But XGBoost differs since it only uses decision trees as its fundamental classifiers [65]. To regulate the complexity of these trees, a modified version of the loss function is employed [66]. The objective function in XGBoost is the sum of a loss function and a regularisation term, as shown in equation 23.

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, yi) + \sum_k \Omega(f_k) \quad (23)$$

where $\mathcal{L}(\phi)$ is the objective function, $l(\hat{y}_i, yi)$ is the loss function and $\Omega(f_k)$ is the regularisation. The regularisation term helps to control the complexity of the model, which helps to prevent overfitting [67]. In XGBoost, the regularisation term is given by Equation (24).

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda\|w\|^2 \quad (24)$$

In the above Equation (24), *T* represents the number of leaves in a tree, and *w* represents the output scores associated with those leaves. By incorporating this loss function into the split criterion of decision trees, a pre-pruning strategy can be implemented [18]. Increasing the value of γ leads to the formation of simpler trees [66]. Essentially, γ determines the minimum amount of loss reduction required to justify splitting an internal node [68].

4. Results and Discussion

This research assessed the effectiveness of four distinct machine learning (ML) methods namely: Lagrangian Support Vector Machines (LSVM), Radial Basis Function Neural Networks (RBFNN), Gaussian Process Regression (GPR) and Extreme Gradient Boosting (XGBoost) in forecasting backbreak at Damang mine. The assessment relied on five performance metrics, namely the correlation coefficient (*r*), coefficient of determination (R^2), mean squared error (MSE), root mean squared error (RMSE) and mean absolute error (MAE). The equations for computing these metrics are as follows:

$$r = \sqrt{1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{(\sum_{i=1}^{n} y_i^2)-(\frac{1}{n}\sum_{i=1}^{n}\hat{y}_i^2)}} \quad (25)$$

$$R^2 = \frac{\Sigma_{i=1}^{n}(y_i-\bar{y}_i)(\hat{y}_i-\bar{\hat{y}}\mathrm{i})}{\sqrt{\sum_{i=1}^{n}(y_i-\bar{y}_i)\Sigma_{i=1}^{n}(\hat{y}_i-\bar{\hat{y}}\mathrm{i})}} \quad (26)$$

$$MSE = \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n} \quad (27)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n}} \quad (28)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i-\hat{y}_i}{y_i}\right| \quad (29)$$

In this equation, *n* represents the total number of samples utilised for training or testing the model, while y_i and $\hat{y}_i$ denote the measured and predicted values, correspondingly.

Considering the results, the XGBoost model exhibited the highest coefficient of determination, suggesting a greater potential to predict backbreak than the other models. Further demonstrating its outstanding prediction performance, the XGBoost model achieved the lowest error among all the models, implying its ability to accurately estimate backbreak in mining operations. Therefore, XGBoost was found to be the best predictor in this investigation based on the assessment measures. It is worth noting that the performance of the models can be affected by the quality and quantity of the data used for training and testing [69-71]. Additionally, while selecting the most appropriate model for a particular application, it is crucial to take into account additional parameters including the computational complexity and the time needed for training and testing [72, 73]. Table 2 presents the performance metrics of different models.

Table 2. *Model performance metrics.*

Model	r	R^2	MSE	RMSE	MAE
LSVM	0.9472	0.8928	0.4222	0.6498	0.4485
RBFNN	0.9632	0.9257	0.3114	0.5581	0.4760
GPR	0.8926	0.7144	0.9234	0.9610	0.5785
XGBoost	0.9788	0.9565	0.1714	0.4139	0.2819

5. Validation, Performance and Comparison of Models

This section introduces a Taylor diagram, scatter plot, and bar chart. These visual aids emphasize the performance of predictive models during testing. Invented by Karl E. Taylor in 1994, the Taylor diagram is a handy tool that quantifies the correlation between models and a reference point, utilizing the Pearson correlation coefficient (r), the root-mean-square error (RMSE), and the standard deviation [74]. On the Taylor diagram, each model is represented as a single point on a two dimensional (2-D) plot, and the closeness of this point to the reference point signifies an optimal model [75]. Figure 3 displays the Taylor diagram specific to the models developed for the testing datasets in this research. According to this figure, the XGBoost model outperforms the other predictive models in backbreak predictions.

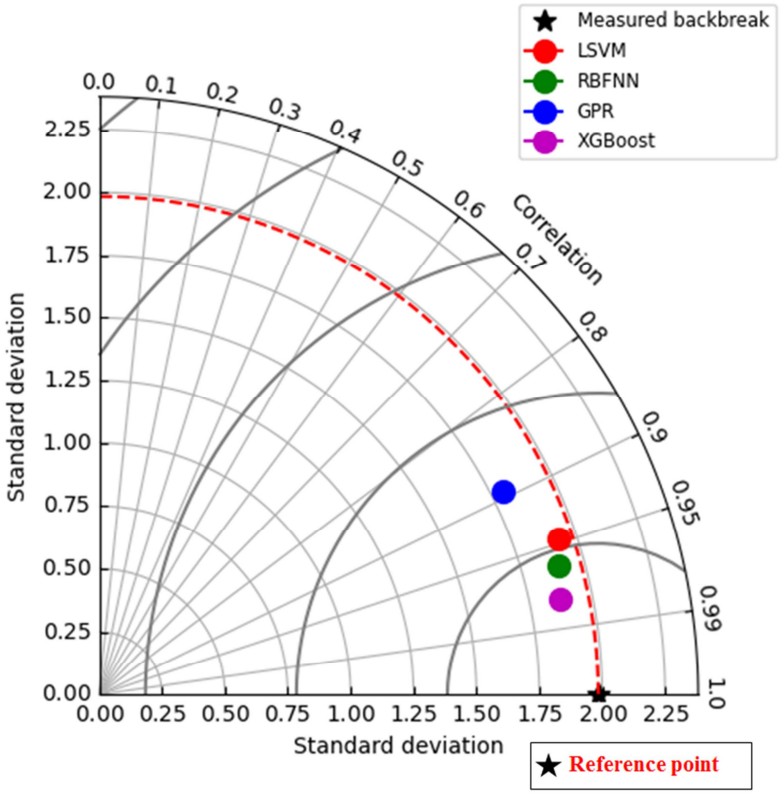

Figure 3. *A Taylor Diagram visualizing the comparative performance of various predictive models.*

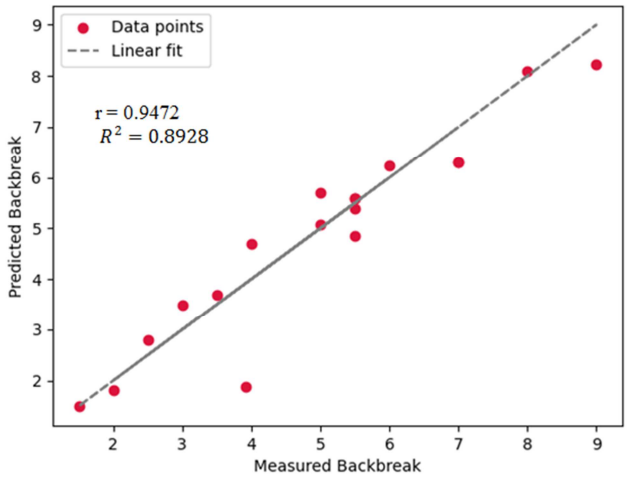

Figure 4. *Measured versus Predicted Backbreak for LSVM.*

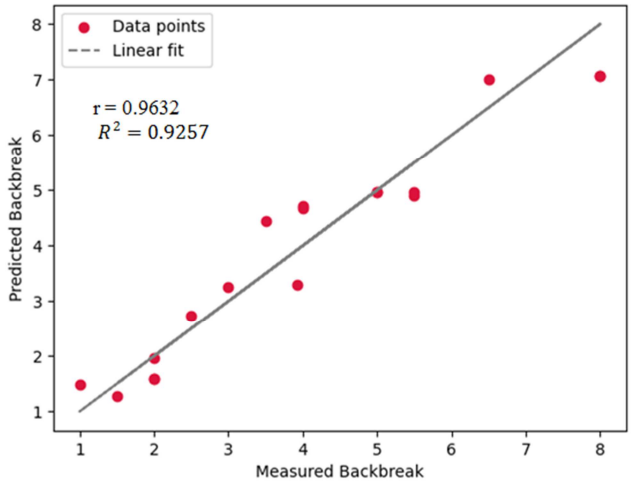

Figure 5. *Measured versus Predicted Backbreak for RBFNN.*

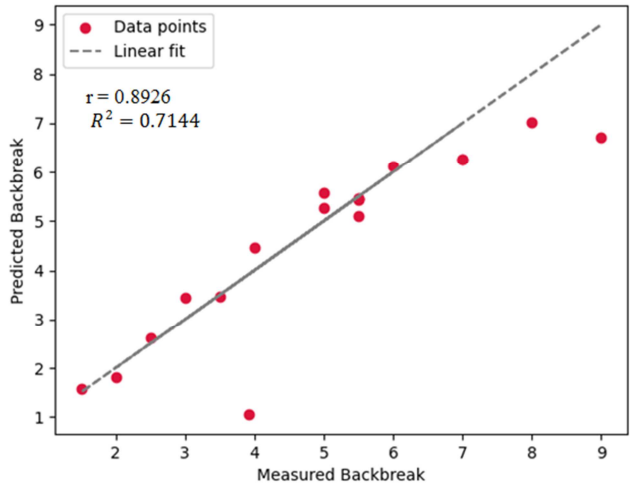

Figure 1. *Measured versus Predicted Backbreak for GPR.*

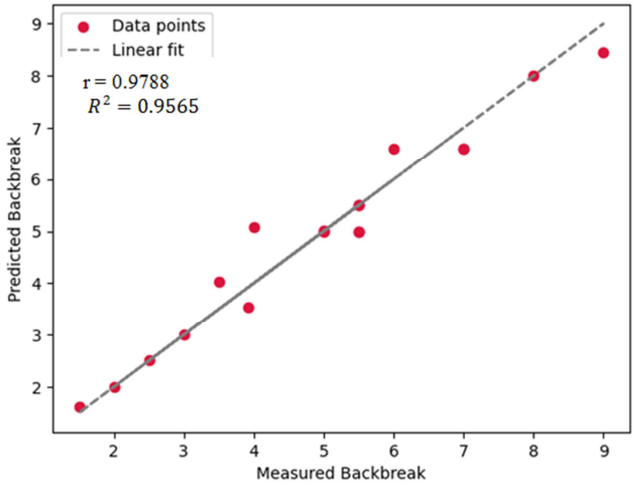

Figure 7. *Measured versus Predicted Backbreak for XGBoost.*

Figure 4 to 7 present the scatter plot of various predicted backbreak of various models against measured backbreak. Evaluating predictive models can be achieved by analyzing the distribution of predicted values. Figure 8 presents bar charts that illustrate the performance of each model, as evaluated by the R-squared metric.

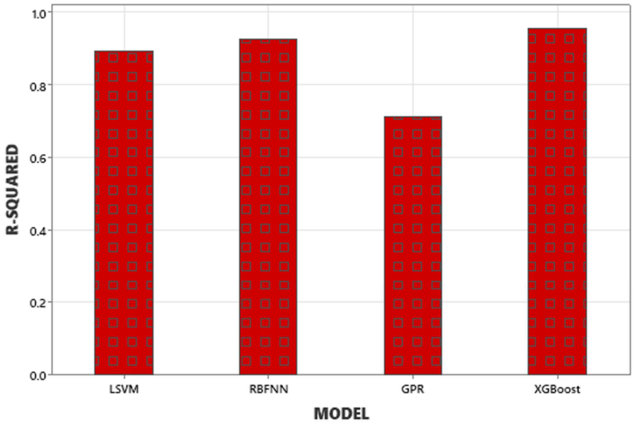

Figure 8. *Comparison of each model with coefficient of determination.*

Sensitivity Analysis

Sensitivity analysis is a widely recognised method used to explore the extent to which a specific variable in a model depends on the various factors present within that model [76]. Performing sensitivity analysis on a comprehensive model can aid in identifying potential areas of concern that may result in strong dependencies between outputs and specific parameters within a dataset [77, 78]. To assess the strength of the relation between backbreak and blasting parameters, the analysis focuses on examining the main effects using simple linear regression (SLR) and the Cosine amplitude method (CAM) of sensitivity analysis. Saeed et al. [79] initially introduced the CAM method to identify the input parameters that have the highest influence on the output parameters. To carry out this technique, a data array X was constructed using all available data pairs in the following manner:

$$X = \{x_1, x_2, \dots, x_n\} \tag{30}$$

In equation (31), each element x_i is a vector of length m, constituting the array X.

$$x_i = \{x_{i1}, x_{i2}, \dots, x_{im}\} \tag{31}$$

In this paper, the model inputs x_i correspond to specific charge [powder factor] (P), geometric stiffness ratio (G), stemming height (T), burden (B), and spacing (S), respectively. The model outputs x_j represent the backbreak. The degree of correlation (r_{ij}) between the model inputs x_i and model outputs x_j is determined by equation (32). The (r_{ij}) values range from zero to one. Higher (r_{ij}) values indicated a greater influence of the input parameter on the output parameter.

$$r_{ij} = \mu_R(X_i, X_j) = \frac{\left|\sum_{k=1}^{m} x_{ik}x_{jk}\right|}{\sqrt{(\sum_{k=1}^{m} x_{ik}^2)(\sum_{k=1}^{m} x_{jk}^2)}} \quad i, j = 1, 2, \dots, n \tag{32}$$

Figures 9 to 14 presents the outcomes derived from the application of simple linear regression. According to the results, powder factor emerged as the most influential parameter in this with a correlation coefficient of 0.8819, indicating its strong relationship with backbreak. On the other hand, the remaining parameters also have significant roles in influencing the severity of backbreak incidents. The findings suggest that as powder factor, spacing and stemming increase, the severity of backbreak tends to increase. Conversely, increasing the geometric stiffness ratio can help mitigate the occurrence of backbreak events. Figure 15 illustrates the findings from utilizing the Cosine Amplitude Method (CAM). This method's results echoed the linear regression outcomes, indicating that the powder factor, with a sensitivity value of $((r_{ij}))$ = 0.9772, was the most sensitive parameter. It was followed by burden $((r_{ij}))$ = 0.9456), stemming $((r_{ij}))$ = 0.9401), spacing $((r_{ij}))$ = 0.9177) and geometric stiffness $((r_{ij}))$ = 0.8505), in that order, based on the conducted sensitivity analysis. The concordance between the findings from both methods suggests that these variables

consistently play pivotal roles in the models regardless of the analytical method applied.

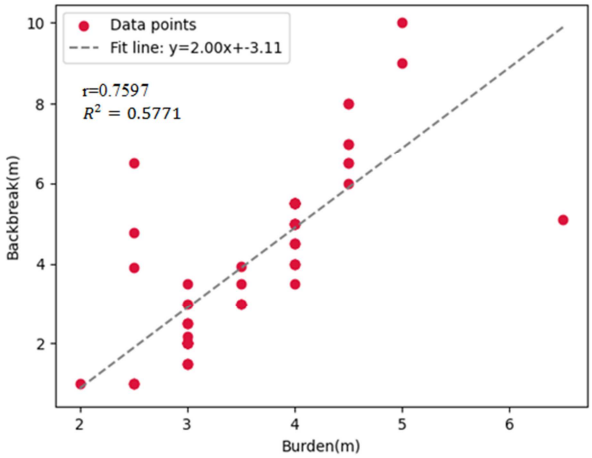

Figure 9. *Correlation of Burden with Backbreak.*

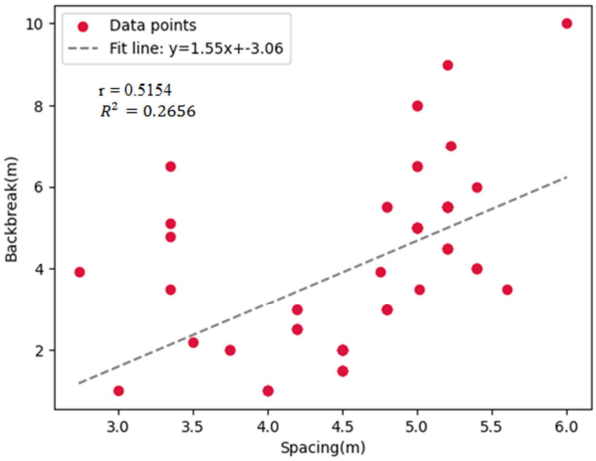

Figure 10. *Correlation of Spacing with Backbreak.*

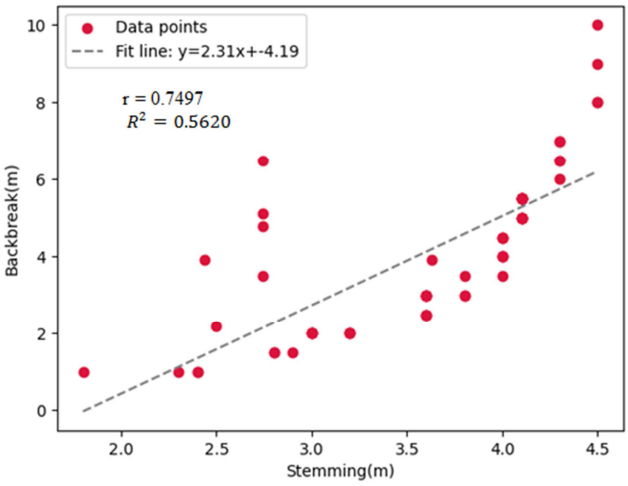

Figure 11. *Correlation of Stemming with Backbreak.*

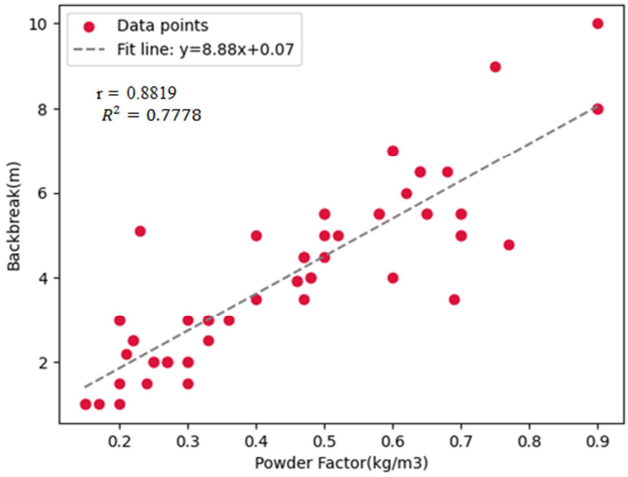

Figure 12. *Correlation of Powder Factor with Backbreak.*

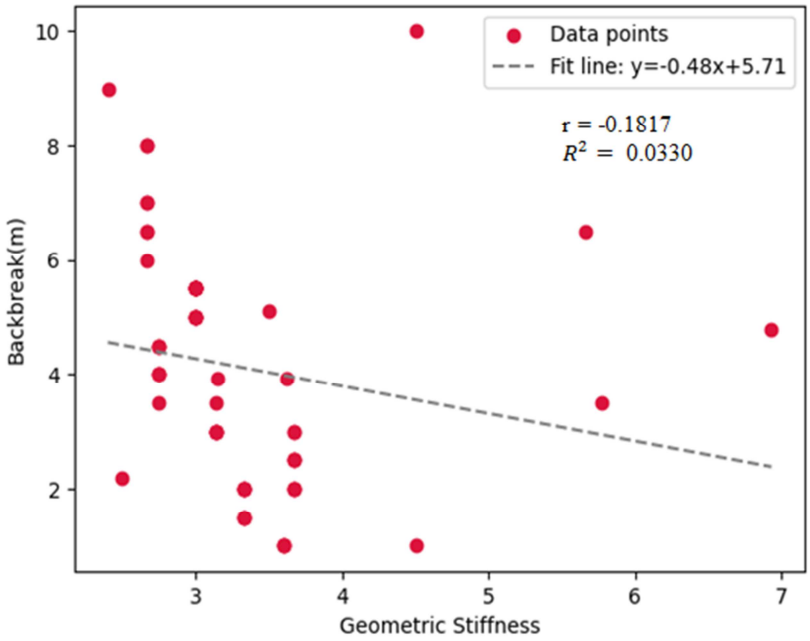

Figure 13. *Correlation of Geometric Stiffness with Backbreak.*

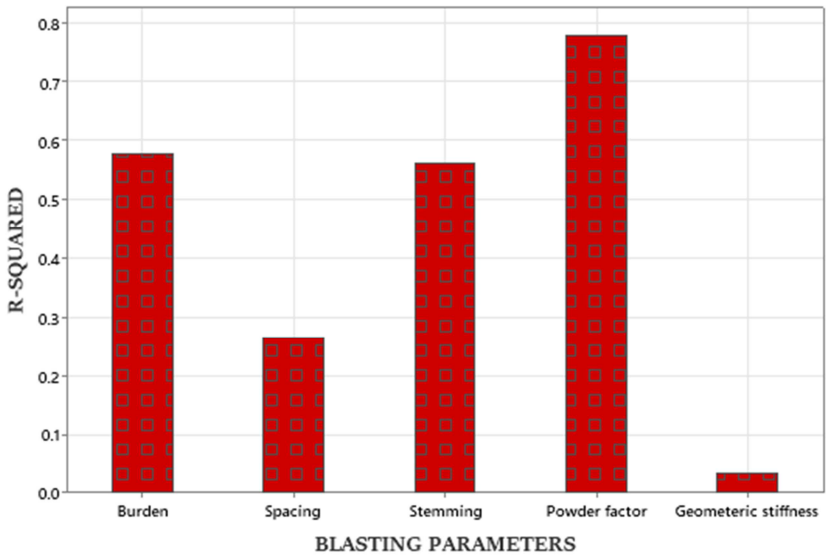

Figure 14. *Sensitivity Analysis for Backbreak using SLR.*

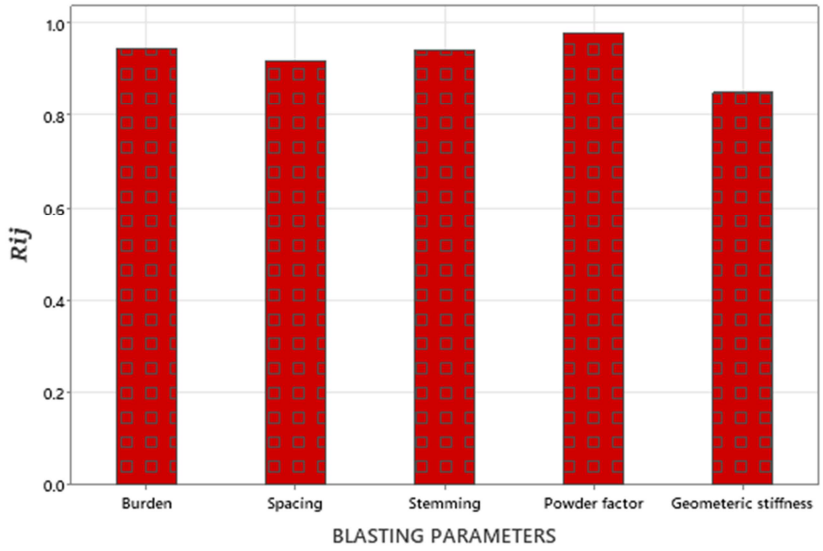

Figure 15. *Sensitivity analysis on backbreak using CAM.*

6. Conclusion and Recommendation

Backbreak is a major safety concern in open pit mines during blasting operations, and accurately predicting its severity is critical for success in both technical and economic aspects. However, predicting backbreak is a highly complex issue that depends on many factors, making it difficult to accurately predict using all relevant parameters. To address this challenge, this study explores the use of four distinct machine learning (ML) techniques: Lagrangian Support Vector Machines (LSVM), Radial Basis Function Neural Networks (RBFNN), Gaussian Process Regression (GPR) and Extreme Gradient Boosting (XGBoost) to determine the most effective predictor of backbreak. Using actual data gathered from 60 blast rounds at the Damang mine, the models were trained and validated. In conclusion, this study demonstrates the potential of machine learning (ML) techniques in predicting backbreak in mining operations. Based on the prediction outcomes, it can be inferred that all four models have exhibited satisfactory performance in predicting backbreak. However, XGBoost outperforms the other three models in terms of effectiveness. Therefore, the XGBoost is highly recommended for backbreak forecasting.

Several recommendations for more research and applications may be made in light of the study's findings. The XGBoost and LSVM models, which have excellent correlation coefficients and relatively low errors, have done well in the prediction of backbreak. As a result, these models may be used in upcoming research and applications including backbreak prediction. Secondly, future research may still investigate the GPR model in greater depth despite its lower effectiveness in prediction compared to the other three models. For example, new hyperparameters may be used to tune the performance of the model or other features can be introduced to the dataset to improve the performance of the GPR model. Thirdly, more

research may be done to assess how well XGBoost, LSVM, RBFNN and GPR models perform in comparison to other regression models like linear regression, decision trees, and random forests. This research can aid in determining the best model for backbreak prediction in various settings. Lastly, these models may be integrated into a decision support system for mining operations backbreak prediction. Such devices can aid in blasting parameter optimisation and backbreak reduction, which can increase mining operations' productivity and safety.

ORCID

Festus Kunkyin-Saadaari: 0000-0002-8202-2021
Victor Kwaku Agadzie: 0009-0000-3452-4150
Richard Gyebuni: 0000-0002-9957-5351

Ethical Statement

The authors state that the research was conducted according to ethical standards.

Acknowledgments

The authors wish to express their profound gratitude to the management of Damang Mine Limited for providing them with data to conduct this research.

Conflicts of Interest

The Authors declare no conflicts of interest.

References

[1] M. Monjezi and H. Dehghani, 'Evaluation of effect of blasting pattern parameters on back break using neural networks', *International Journal of Rock Mechanics and Mining Sciences*, vol. 45, no. 8, pp. 1446–1453, 2008, doi: 10.1016/j.ijrmms.2008.02.007.

[2] I. W. Auma, 'Department of Mining and Water Resources Final Year Project Report Investigating the Causes of Rock Overbreak During Blasting Operations', 2016.

[3] S. Bhandari, *Engineering_rock_blasting_operations_bha.pdf*. Taylor & Francis, 1997. [Online]. Available: https://books.google.com.gh/books?id=ZOtyQgAACAAJ

[4] P.-A. Persson, H. Roger, and L. Jaimin, *Pdfcoffee.Com_Rock-Blasting-and-Explosives-Engineering-2-Pdf-Free*. Taylor & Francis, 1993.

[5] W. Hustrulid, 'Blasting principles for open pit mining: Volume 1 - General design concepts', vol. 1. p. 412, 1999.

[6] C. J. Konya and E. J. Walter, 'Rock blasting and overbreak control (No. FHWA-HI-92-001; NHI-13211)', *National Highway Institute*, no. 132, p. 430, 1991.

[7] W. C. B. Gates, L. T. Ortiz, and R. M. Florez, 'Analysis of rockfall and blasting backbreak problems, US 550, Molas Pass, CO', *American Rock Mechanics Association - 40th US Rock Mechanics Symposium, ALASKA ROCKS 2005: Rock Mechanics for Energy, Mineral and Infrastructure Development in the Northern Regions*, pp. 671–677, 2005, [Online]. Available: https://onepetro.org/ARMAUSRMS/proceedings-pdf/ARMA05/All-ARMA05/ARMA-05-671/1841977/arma-05-671.pdf

[8] K. & E. Walter, *Surface Blast Design_Calvin J. Konya & Edward J. Walter_1990*. 1990.

[9] D.-D. N. K. Bansah K. J., Arko-Gyimah K, Kansake B. A., 'Mitigating Blast Vibration Impact', *4th UMaT Biennial International Mining and Mineral Conference*, no. August, pp. 30–36, 2016.

[10] P. Ragam and D. S. Nimaje, 'Evaluation and prediction of blast-induced peak particle velocity using artificial neural network: A case study', *Noise and Vibration Worldwide*, vol. 49, no. 3, pp. 111–119, 2018, doi: 10.1177/0957456518763161.

[11] R. Kumar, D. Choudhury, and K. Bhargava, 'Determination of blast-induced ground vibration equations for rocks using mechanical and geological properties', *Journal of Rock Mechanics and Geotechnical Engineering*, vol. 8, no. 3, pp. 341–349, 2016, doi: 10.1016/j.jrmge.2015.10.009.

[12] R. J. Gagnon, 'Empirical Research: The Burdens and the Benefits', *Interfaces (Providence)*, vol. 12, no. 4, pp. 98–102, 1982, doi: 10.1287/inte.12.4.98.

[13] S. H. Sahir, K. Minan, S. Samsudin, I. Zufria, and R. Rahim, 'Adaptive neuro fuzzy inference system for prediction: A study approach', *International Journal of Engineering and Technology (UAE)*, vol. 7, no. 2.14 Special Issue 14, pp. 260–263, 2018, doi: 10.14419/ijet.v7i1.1.9482.

[14] E. Ebrahimi, M. Monjezi, M. R. Khalesi, and D. J. Armaghani, 'Prediction and optimization of back-break and rock fragmentation using an artificial neural network and a bee colony algorithm', *Bulletin of Engineering Geology and the Environment*, vol. 75, no. 1, pp. 27–36, 2015, doi: 10.1007/s10064-015-0720-2.

[15] A. A. Bazzazi and M. Esmaeili, 'Prediction of backbreak in open pit blasting by adaptive neuro-fuzzy inference system', *Archives of Mining Sciences*, vol. 57, no. 4, pp. 933–943, 2012, doi: 10.2478/v10267-012-0062-x.

[16] Q. Yu, M. Monjezi, A. S. Mohammed, H. Dehghani, D. J. Armaghani, and D. V. Ulrikh, 'Optimized support vector machines combined with evolutionary random forest for prediction of back-break caused by blasting operation', *Sustainability (Switzerland)*, vol. 13, no. 22, 2021, doi: 10.3390/su132212797.

[17] R. Shirani Faradonbeh, M. Monjezi, and D. Jahed Armaghani, 'Genetic programing and non-linear multiple regression techniques to predict backbreak in blasting operation', *Eng Comput*, vol. 32, no. 1, pp. 123–133, 2016, doi: 10.1007/s00366-015-0404-3.

[18] Z. Nabavi, M. Mirzehi, H. Dehghani, and P. Ashtari, 'A Hybrid Model for Back-Break Prediction using XGBoost Machine learning and Metaheuristic Algorithms in Chadormalu Iron Mine', *Journal of Mining and Environment*, vol. 14, no. 2, pp. 689–712, 2023, doi: 10.22044/jme.2023.12796.2323.

[19] C. K. Arthur, V. A. Temeng, and Y. Y. Ziggah, 'Novel approach to predicting blast-induced ground vibration using Gaussian process regression', *Eng Comput*, vol. 36, no. 1, pp. 29–42, 2020, doi: 10.1007/s00366-018-0686-3.

[20] J. Gareth, W. Daniela, T. Hastie, and T. Robert, 'An Introduction to Statistical Learning with Applications in R', *Springer Science & Business Media*, 2013. https://books.google.com.gh/books?id=qcI%5C_AAAAQBAJ%7D,%0A year=%7B2013

[21] O. A. Montesinos López, A. Montesinos López, and C. Jose, *Overfitting, Model Tuning, and Evaluation of Prediction Performance*. Springer, Cham, 2022. [Online]. Available: https://doi.org/10.1007/978-3-030-89010-0_4

[22] M. Kuhn and K. Johnson, *Applied predictive modeling*. 2013. doi: 10.1007/978-1-4614-6849-3.

[23] V. K. Paspula, 'Splitting a data set into training, validation and test sets', 2020. https://vinaypaspula.substack.com/p/splitting-a-data-set-into-training-validation-and-test-sets-f1654b7574c (accessed Jul. 28, 2023).

[24] J. Brownlee, 'Train-Test Split for Evaluating Machine Learning Algorithms', *Machine Learning Mastery*, 2020. https://machinelearningmastery.com/train-test-split-for-evaluating-machine-learning-algorithms/ (accessed Oct. 28, 2023).

[25] M. Behzad, K. Asghari, M. Eazi, and M. Palhang, 'Generalization performance of support vector machines and neural networks in runoff modeling', *Expert Syst Appl*, vol. 36, no. 4, pp. 7624–7629, 2009, doi: 10.1016/j.eswa.2008.09.053.

[26] V. N. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed., no. 1. 1999. doi: https://doi.org/10.1007/978-1-4757-3264-1.

[27] N. Sapankevych and R. Sankar, 'Time series prediction using support vector machines: A survey', *IEEE Comput Intell Mag*, vol. 4, no. 2, pp. 24–38, 2009, doi: 10.1109/MCI.2009.932254.

[28] H. Guo, H. Nguyen, X. N. Bui, and D. J. Armaghani, 'A new technique to predict fly-rock in bench blasting based on an ensemble of support vector regression and GLMNET', *Eng Comput*, vol. 37, no. 1, pp. 421–435, 2021, doi: 10.1007/s00366-019-00833-x.

[29] S. Mukherjee, E. Osuna, and F. Girosi, 'Nonlinear prediction of chaotic time series using support vector machines', *Neural Networks for Signal Processing - Proceedings of the IEEE Workshop*, no. July, pp. 511–520, 1997, doi: 10.1109/nnsp.1997.622433.

[30] J. T. Jeng, C. C. Chuang, and S. F. Su, 'Support vector interval regression networks for interval regression analysis', *Fuzzy Sets Syst*, vol. 138, no. 2, pp. 283–300, 2003, doi: 10.1016/S0165-0114(02)00570-5.

[31] M. Sabzekar and S. M. H. Hasheminejad, 'Robust regression using support vector regressions', *Chaos Solitons Fractals*, vol. 144, p. 110738, 2021, doi: 10.1016/j.chaos.2021.110738.

[32] V. D. A. Sánchez, 'Advanced support vector machines and kernel methods', *Neurocomputing*, vol. 55, no. 1–2, pp. 5–20, 2003, doi: 10.1016/S0925-2312(03)00373-4.

[33] Q. A. Tran, X. Li, and H. Duan, 'Efficient performance estimate for one-class support vector machine', *Pattern Recognit Lett*, vol. 26, no. 8, pp. 1174–1182, 2005, doi: 10.1016/j.patrec.2004.11.001.

[34] C. Wikipedia, 'Bra – ket notation', *Wikipedia, The Free Encyclopedia.*, 2023.

[35] H. J. Lin and J. P. Yeh, 'Optimal reduction of solutions for support vector machines', *Appl Math Comput*, vol. 214, no. 2, pp. 329–335, 2009, doi: 10.1016/j.amc.2009.04.010.

[36] B. Schölkopf, A. Smola, and K. R. Müller, 'Nonlinear Component Analysis as a Kernel Eigenvalue Problem', *Neural Comput*, vol. 10, no. 5, pp. 1299–1319, 1998, doi: 10.1162/089976698300017467.

[37] Y. G. Wang, X. Lin, and M. Zhu, 'Robust estimating functions and bias correction for longitudinal data analysis', *Biometrics*, vol. 61, no. 3, pp. 684–691, 2005, doi: 10.1111/j.1541-0420.2005.00354.x.

[38] C. Savas and F. Dovis, 'The impact of different kernel functions on the performance of scintillation detection based on support vector machines', *Sensors (Switzerland)*, vol. 19, no. 23, pp. 1–16, 2019, doi: 10.3390/s19235219.

[39] A. Ben-Hur and J. Weston, 'A user's guide to support vector machines.', *Methods Mol Biol*, vol. 609, no. January 2010, pp. 223–239, 2010, doi: 10.1007/978-1-60327-241-4_13.

[40] K. B. Kim, J. B. Park, Y. H. Choi, and G. Chen, 'Control of chaotic dynamical systems using radial basis function network approximators', *Inf Sci (N Y)*, vol. 130, no. 1–4, pp. 165–183, 2000, doi: 10.1016/S0020-0255(00)00074-8.

[41] C. S. K. Dash, A. K. Behera, S. Dehuri, and S. B. Cho, 'Radial basis function neural networks: A topical state-of-the-art survey', *Open Computer Science*, vol. 6, no. 1, pp. 33–63, 2016, doi: 10.1515/comp-2016-0005.

[42] T. Poggio and F. Girosi, 'Networks for Approximation arid Learning', *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1481–1497, 1990, doi: 10.1109/5.58326.

[43] C. Bishop, 'Improving the Generalization Properties of Radial Basis Function Neural Networks', *Neural Comput*, vol. 3, no. 4, pp. 579–588, 1991, doi: 10.1162/neco.1991.3.4.579.

[44] S. Haykin, J. Nie, and B. Currie, 'Neural network-based receiver for wireless communications', *Electronics Letters*, vol. 35, no. 3. pp. 203–205, 1999. doi: 10.1049/el:19990177.

[45] M. J. L. Orr, 'Introduction to radial basis function networks', *University of Edinburgh*, pp. 1–67, 1996.

[46] C. Bishop, *Pattern Recognition and Machine Learning*. 2006. doi: 10.4324/9780203733332.

[47] B. Schölkopf and A. J. Smola, 'Support Vector Machines and Kernel Algorithms', *The Handbook of Brain Theory and Neural Networks*, pp. 1119–1125, 2002.

[48] J. Park and I. W. Sandberg, 'Universal Approximation Using Radial-Basis-Function Networks', *Neural Comput*, vol. 3, no. 2, pp. 246–257, 1991, doi: 10.1162/neco.1991.3.2.246.

[49] S. Chen, C. F. N. Cowan, and P. M. Grant. 'Orthogonal least squares learning algorithm for radial basis function networks'. *IEEE Transactions on Neural Networks, Vol.* 2, No. 2, pp. 302-309, 1991.

[50] C. E. Rasmussen, 'Gaussian Processes in machine learning', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3176, no. September 2004, pp. 63–71, 2004, doi: 10.1007/978-3-540-28650-9_4.

[51] M. Ebden, 'Gaussian Processes: A Quick Introduction', no. May 2015, 2015, [Online]. Available: http://arxiv.org/abs/1505.02965

[52] N. A. C. Cressie, *statistics for spatial*, 2nd ed. New York, Chichester, Toronto, Brisbane, Singapore: A Wiley-Interscience Publication, 2015.

[53] T. Beckers, 'An Introduction to Gaussian Process Models', 2021, [Online]. Available: http://arxiv.org/abs/2102.05497

[54] J. Franklin, 'The elements of statistical learning: data mining, inference and prediction', *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005, doi: 10.1007/bf02985802.

[55] J. J. Li, A. Jutzeler, and B. Faltings, 'Estimating urban ultrafine particle distributions with Gaussian process models', *CEUR Workshop Proc*, vol. 1142, no. April, pp. 145–153, 2014.

[56] L. Yan, 'Covariance'. pp. 1–2, 2019.

[57] J. Vojtassak, P. Maresch, F. Makai, and J. Tkacik, 'Analysis of Covariance', *Rheumatologia*, vol. 11, no. 1, pp. 53–55, 1997.

[58] D. Pollard, 'Variances and covariances', no. September. pp. 1–6, 1997.

[59] E. L. Snelson, 'Flexible and efficient Gaussian process models for machine learning', *ACM SIGKDD Explorations Newsletter*, vol. 7, no. 2001, pp. 1–135, 2007, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.62.4041&rep=rep1&type=pdf%5Cnhttp://portal.acm.org/citation.cfm?id=1117456

[60] J. Liu, K. Z. Yan, X. Zhao, and Y. Hu, 'Prediction of autogenous shrinkage of concretes by support vector machine', *International Journal of Pavement Research and Technology*, vol. 9, no. 3, pp. 169–177, 2016, doi: 10.1016/j.ijprt.2016.06.003.

[61] C. J. Moore, A. J. K. Chua, C. P. L. Berry, and J. R. Gair, 'Fast methods for training gaussian processes on large datasets', *R Soc Open Sci*, vol. 3, no. 5, pp. 0–9, 2016, doi: 10.1098/rsos.160125.

[62] A. V. Mueller and H. F. Hemond, 'Extended artificial neural networks: Incorporation of a priori chemical knowledge enables use of ion selective electrodes for in-situ measurement of ions at environmentally relevant levels', *Talanta*, vol. 117, pp. 112–118, 2013, doi: 10.1016/j.talanta.2013.08.045.

[63] S. Malik, R. Harode, and A. Singh Kunwar, 'XGBoost: a deep dive into boosting', *Simon Fraser University*, no. February, pp. 1–21, 2020, doi: 10.13140/RG.2.2.15243.64803.

[64] B. Candice, C. Anna, and M. Gonzalo. 'A Comparative Analysis of XGBoost', 2019.

[65] J. H. Friedman, 'Greedy Function Approximation: A Gradient Boosting Machine', *Institute of Mathematical Statistics*, vol. 29, no. 5, pp. 1189–1232, 2014, [Online]. Available: http://www.jstor.org/stable/2699986

[66] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, 'A Comparative Analysis of XGBoost', no. November 2019, 2019, doi: 10.1007/s10462-020-09896-5.

[67] R. Kesarwani, 'XGBoost: A BOOSTING Ensemble', 2021. https://medium.com/almabetter/xgboost-a-boosting-ensemble-b273a71de7a8 (accessed Oct. 28, 2023).

[68] T. Chen and C. Guestrin, 'XGBoost: A scalable tree boosting system', *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. 13-17-August-2016, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.

[69] J. Heaton, 'Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning', *Genet Program Evolvable Mach*, vol. 19, no. 1–2, pp. 305–307, 2018, doi: 10.1007/s10710-017-9314-z.

[70] P. Domingos, 'A few useful things to know about machine learning', *Commun ACM*, vol. 55, no. 10, pp. 78–87, 2012, doi: 10.1145/2347736.2347755.

[71] G. Varoquaux, P. R. Raamana, D. A. Engemann, A. Hoyos-Idrobo, Y. Schwartz, and B. Thirion, 'Assessing and tuning brain decoders: Cross-validation, caveats, and guidelines', *Neuroimage*, vol. 145, no. June, pp. 166–179, 2017, doi: 10.1016/j.neuroimage.2016.10.038.

[72] H. Zhang, 'The optimality of Naive Bayes', *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2004*, vol. 2, pp. 562–567, 2004.

[73] S. Raschka, 'Naive Bayes and Text Classification I - Introduction and Theory', pp. 1–20, 2014.

[74] H. Fisher *et al.*, 'Taylor_Diagram.Pdf', 2023. https://metplotpy.readthedocs.io/en/latest/Users_Guide/index.html (accessed Jul. 29, 2023).

[75] K. E. Taylor, 'in a Single Diagram', vol. 106, pp. 7183–7192, 2001.

[76] F. Pianosi *et al.*, 'Sensitivity analysis of environmental models: A systematic review with practical workflow', *Environmental Modelling and Software*, vol. 79, no. May, pp. 214–232, 2016, doi: 10.1016/j.envsoft.2016.02.008.

[77] J. C. Helton, J. D. Johnson, C. J. Sallaberry, and C. B. Storlie, 'Survey of sampling-based methods for uncertainty and sensitivity analysis', *Reliab Eng Syst Saf*, vol. 91, no. 10–11, pp. 1175–1209, 2006, doi: 10.1016/j.ress.2005.11.017.

[78] I. M. Sobol, 'Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates', *Math Comput Simul*, vol. 55, no. 1–3, pp. 271–280, 2001, doi: 10.1016/S0378-4754(00)00270-6.

[79] M. S. Saeed, H. Jalalifar, H. Shamsoddini, and M. Darbor, 'A comparative study on the application of Regression-PSO and ANN methods to predict backbreak in open-pit mines A comparative study on the application of Regression-PSO and ANN methods to predict backbreak in open-pit mines', vol. 11, no. June, pp. 55–66, 2022, doi: 10.29252/ANM.2022.16933.1508.